#Task-1 Write a query to fetch details of all completed appointments, including the patient’s name, doctor’s name, and specialization.

SELECT

a.appointment\_id,

p.name AS patient\_name,

d.name AS doctor\_name,

d.specialization,

a.appointment\_date,

a.status

FROM appointments a

JOIN patients p ON a.patient\_id = p.patient\_id

JOIN doctors d ON a.doctor\_id = d.doctor\_id

WHERE a.status = 'Completed';

#Task-2 Retrieve all patients who have never had an appointment. Include their name, contact details, and address in the output.

SELECT

p.patient\_id,

p.name as patient\_name,

p.contact\_number,

p.address

from patients as p

left join appointments as a on p.patient\_id=a.patient\_id

where appointment\_id is null;

#Task-3 Find the total number of diagnoses for each doctor, including doctors who haven’t diagnosed any patients.

#Display the doctor’s name, specialization, and total diagnoses.

SELECT

do.name as doctorsname,do. doctor\_id,

do.specialization,

count(d.diagnosis\_id) as totaldiagnosis

from diagnoses as d

right join doctors as do on do.doctor\_id=d.doctor\_id

group by doctorsname,specialization,doctor\_id

order by totaldiagnosis desc;

#Task-4 Write a query to identify mismatches between the appointments and diagnoses tables.

#Include all appointments and diagnoses with their corresponding patient and doctor details.

SELECT

a.appointment\_id,

a.patient\_id,

p.name as patientname,

a.doctor\_id,

d.name as doctorname,

a.appointment\_date,

diag.diagnosis\_id,

diag.diagnosis,

diag.diagnosis\_date

FROM appointments a

LEFT JOIN diagnoses diag

ON a.patient\_id = diag.patient\_id and a.doctor\_id=diag.doctor\_id

LEFT JOIN patients p

ON a.patient\_id = p.patient\_id

LEFT JOIN doctors d

ON a.doctor\_id = d.doctor\_id

WHERE diag.diagnosis\_id IS NULL

OR a.appointment\_date <> diag.diagnosis\_date;

# Task- 5 For each doctor, rank their patients based on the number of appointments in descending order.

select

a.doctor\_id,

d.name as doctorname,

a.patient\_id,

p.name as patientname,

count(a.appointment\_id) as totalappointments,

rank() over(partition by a.doctor\_id order by count(a.appointment\_id) desc) as patientrank

from appointments as a

join patients as p on p.patient\_id=a.patient\_id

join doctors as d on d.doctor\_id=a.doctor\_id

group by doctor\_id,doctorname,patient\_id,patientname

order by doctor\_id,patientrank desc;

#Task-6 Write a query to categorize patients by age group (e.g., 18-30, 31-50, 51+).

#Count the number of patients in each age group.

SELECT

CASE

WHEN age BETWEEN 18 AND 30 THEN '18-30'

WHEN age BETWEEN 31 AND 50 THEN '31-50'

WHEN age >= 51 THEN '51+'

ELSE 'Unknown'

END AS age\_group,

COUNT(\*) AS patient\_count

FROM (

SELECT patient\_id,

age

FROM patients

) as patientage

GROUP BY age\_group

ORDER BY age\_group;

#Task-7 Retrieve a list of patients whose contact numbers end with "1234" and display their names in uppercase.

select upper(name) as patientname,contact\_number

from patients

where contact\_number Like '%1234';

#Task-8 Find patients who have only been prescribed "Insulin" in any of their diagnoses.

SELECT p.patient\_id, p.name as patientname,m.medication\_name

FROM patients p

JOIN diagnoses d ON p.patient\_id = d.patient\_id

JOIN medications m ON d.diagnosis\_id = m.diagnosis\_id

GROUP BY patient\_id, patientname,medication\_name

HAVING COUNT(DISTINCT m.medication\_name) = 1 AND MAX(m.medication\_name) = 'Insulin';

#Task-9 Calculate the average duration (in days) for which medications are prescribed for each diagnosis.

select d.diagnosis,avg(datediff (m.start\_date, m.end\_date)) as avgduration

from diagnoses as d

join medications as m

on d.diagnosis\_id=m.diagnosis\_id

WHERE m.end\_date IS NOT NULL AND m.start\_date IS NOT NULL

group by d.diagnosis

order by avgduration desc;

# Task-10 Write a query to identify the doctor who has attended the most unique patients.

#Include the doctor’s name, specialization, and the count of unique patients.

select d.name as doctorname, d.specialization,d.doctor\_id,

count(distinct p.patient\_id) as uniquepatientname

from patients as p

join appointments as a on a.patient\_id=p.patient\_id

join doctors as d on d.doctor\_id=a.doctor\_id

group by d.doctor\_id,doctorname,d.specialization

order by uniquepatientname desc;